UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/058,050	01/29/2002	David R. Blythe	1026.00	3659

26111 7590 08/13/2007
STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 NEW YORK AVENUE, N.W.
WASHINGTON, DC 20005

EXAMINER	
LAY, MICHELLE K	
ART UNIT	**PAPER NUMBER**
2628	
MAIL DATE	**DELIVERY MODE**
08/13/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)
	10/058,050	BLYTHE ET AL.
	Examiner	Art Unit
	Michelle K. Lay	2628

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *24 July 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-5 and 8-21* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-5 and 8-21* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *20070724*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendment

The amendment filed 07/24/2002 has been entered and made of record. Claims 1-5 and 8-21 are pending.

Response to Arguments

Applicant's arguments filed 07/24/2007 have been fully considered but they are not persuasive. Applicant remarks the Examiner interprets "[a] scene, or portions of a scene, can be divided into pixel regions" as teaching *both* a geometry chunk and a subarea of a compositing window. Examiner respectfully disagrees. Kenworthy et al. (5,808,617) teaches a scene, or portions of a scene, can be divided into pixel regions called chunks [c.8 L 33-34]. These pixel regions correspond to Applicant's *subarea of a compositing window.* Furthermore, Kenworthy teaches the system divides the geometry (said *geometry chunk*) assigned to gsprites into chunks [c.8 L35-41], meaning that the system of Kenworthy divides the geometry according to the pixel regions they reside in. Therefore, Kenworthy teaches both a geometry chunk and a subarea as separate entities.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims **1-5** and **8-21** are rejected under 35 U.S.C. 103(a) as being unpatentable over Kenworthy et al. (5,808,617) in view of Bowen et al. (6,292,200 B1).

Kenworthy teaches the limitations of claims 1-5 and 8-21 with the exception of explicitly disclosing a graphics pipeline for parallel processing. However, Bowen teaches a computer graphics system having multiple rendering pipes to process graphics data at the same time [*abstract*].

In regards to claim **1**, Kenworthy teaches *a method for minimizing an amount of data needed to test a geometry chunk in a frame against subarea boundaries in a compositing window, comprising the steps of:*

- ***defining the geometry chunk with a bounding region, wherein said bounding region defines a space the geometry chunk occupies on the compositing window;***

  A scene, or portions of a scene, can be divided into pixel regions (e.g. 32x32 pixels), called chunks (said ***boundary region***). The geometry is presorted into bins based in which chunk the geometry will be rendered into [col. 8, lines 32-39], meaning that the system of Kenworthy divides the geometry (said ***geometry chunk***) according to the pixel regions they reside in.

- ***storing data that defines said bounding region for use in processing the geometry chunk in a subsequent frame;***

  Main memory (134) [Fig. 2].

- ***sending said data that defines said bounding region to graphics pipelines;***

  Gsprite chunk data is sent to the buffer [col. 12, lines 27-30].

Although not explicitly taught, it is implicit that the chunk boundary is communicated through the system via the memory control (136) that serves as an interface between the processor (132) and main memory (134) as shown in Fig. 2. The geometry is presorted into bins based on which chunk the geometry will be rendered into [col. 8, lines 32-39].

- ***determining, from said data that defines said bounding region, a graphics pipeline of said graphics pipelines that will render the geometry chunk;***

  The geometry is presorted into bins based on which chunk the geometry will be rendered into [col. 8, lines 32-39; col. 9, line 27].

- ***assigning a subarea in the compositing window to receive an output of said graphics pipeline; and***

  A scene, or portions of a scene, can be divided into pixel regions, called chunks [col. 8, lines 32-39]. These regions

- ***communicating data associated with the geometry chunk to said graphics pipeline;***

  The geometry is presorted into bins based on which chunk the geometry will be rendered into [col. 8, lines 32-39]. The image processor determines how the geometric primitives (e.g. polygons) should be divided among the chunks

[col. 13, lines 55-60]. Although not explicitly taught, it is implicit that such information is communicated through the system via the system interface (110) as shown in Fig. 1.

- ***wherein said graphics pipelines are configured to render the frame by spatial compositing through parallel processing, said data that defines said bounding region is less than said data associated with the geometry chunk, and the geometry chunk is different from said subarea.*** The system of Kenworthy further teaches the use of double buffering, which enables the system to generate one display list while it reads another. As the system calculates the gsprite transforms and build the display list for one frame, it reads the display list for another frame and displays the image data in this list. Because of the double buffering, the image preprocessor performs steps (280-286) shown in Fig. 6, for one frame, which the image processor performs steps (290-298) for another frame [col. 15, lines 23-32].

Bowen teaches a method/system having a hyperpipe architecture for utilizing multiple rendering pipes for the generation of a single 3-D display [col. 3 lines 47-49]. The application program running on host processor (H) (301) [Fig. 3] provides the high-level instructions and data to be used in the rendering process. The information is passed on to a geometry engine (G) (302), which performs the arithmetic operations on vertices. Multiple pipes are merged together to help in rendering a single frame, thereby allowing parallel processing of complex images [col. 5 lines 41-43]. The appropriate pixel values are read from frame buffer (305) by display block (D) (304) and put out onto the

hyperpipe bus or drawn out for display onto a CRT screen [col. 6 lines 5-8]. With reference to Fig. 5, a rendering pipe may be instructed to contribute in the rendering of a portion of a frame. The portion of the frame is specified according to an XY coordinate system. Registers (504) and (505) store the results from the rendering pipes (Pipes 0, 1). Data is then merged and passed to an output device. Note that the frame can have separate sections rendered by different rendering pipes. As an example, for a two rendering pipe system, the display surface (512) is subdivided into four sections. Pipe 0 renters two sections, and pipe 1 renders two sections [col. 6 lines 35-59].

Therefore, it would have been obvious to one of ordinary skill in the art to utilize the multiple pipeline architecture of Bowen with the method/system of Kenworthy in order to eliminate the double buffering of Kenworthy and minimize the amount of time to render an image

In regards to claims **2**, **10**, and **18**, Bowen teaches a hyperpipe router (201) [Fig. 2], which determines which packet is intend for which pipe. The packed is then routed to a local router (202) that directs the packet to the appropriate circuit within pipe (e.g., the rasterizer) [col. 5 lines 46-60].

In regards to claim **2**, the image processor determines how the geometric primitives (e.g. polygons) should be divided among the chunks [col. 13, lines 55-60].

In regards to claim **3**, Kenworthy teaches ***wherein said space is a screen space*** [col. 13, lines 50-54]. Additionally, as shown in Fig. 6, the images are read from shared memory (216), and transformed to physical output device coordinates (said ***screen space***) by the gsprite engine (204) [col. 14, lines 25-31; col. 15, lines 44-45].

In regards to claim **4**, Kenworthy teaches ***wherein said space is a world space*** [com. 12, line 50].

In regards to claim **5**, Kenworthy teaches ***wherein said space is an object space.***

> The invention of Kenworthy transforms the bounding volume so that the number of chunks required to render the gsprite is minimized. Therefore, the space to which the objects assigned to the gsprite is not necessarily the screen space but referred to the gsprite space [col. 13, lines 47-54].

In regards to claim **8**, Kenworthy teaches a display list that defines which gsprites (i.e. chunks) are to be displayed on the screen [col. 14, lines 45-50].

In regards to claim **9**, Kenworthy teaches geometry processing where each primitive has a series of vertices. The vertex includes position information. Although Kenworthy does not explicitly teach such vertices as the vertices associated with the chunk, the chunk consists of the primitives associated with the vertices and therefore, implicitly, the geometry chunk is represented as a vertex array object. Additionally, the system of

Kenworthy consists of a vertex input processor (384), vertex and control registers (386) which is used for edge detections of the triangle [col. 17, lines 19-24].

In regards to claim **10**, Kenworthy teaches the image preprocessor determines how the geometric primitives (e.g. polygons) should be divided among the chunks by transforms the polygons to 2-D space and determining which chunk or chunks the polygons project into. Due to the expense of clipping polygons, the preferred approach is to not clip the polygons lying at the edge of a chunk. Instead, a chunk includes polygons that overlap its edge. If a polygon extends over the border of two chunks, for example, the vertices of the polygon are included in each chunk. The image preprocessor then queues the chunk data for tiling. Tiling refers to the process of determining pixel values such as color and alpha for pixel locations covered or partially covered by one or more polygons [col. 13, line 55 – col. 14, line 2]. The tiler includes registers for six vertices to allow double buffering of the triangle processing [col. 17, lines 19-27].

In regards to claim **11**, claim 11 recites the same limitations as claim 1. Therefore, the same rationale used for claim 1 is applied. Furthermore, system of Kenworthy provides a graphics-rendering pipeline [*Abstract*]. The graphics support software (160) includes functions to support chunking and gsprite allocation (said ***geometry distributor***) [col. 11, lines 21-40]. The graphics support software (160) executes on the host computer system (130) and communicates with the image processing board

(164) through the hardware abstraction layer (162) (said ***interface***) [col. 11, lines 14-20]. The image processor stores the compressed chunk in shared memory (262) [col. 15, lines 53-67]. Furthermore, the system of Kenworthy further teaches the use of double buffering, which enables the system to generate one display list while it reads another. As the system calculates the gsprite transforms and build the display list for one frame, it reads the display list for another frame and displays the image data in this list. Because of the double buffering, the image preprocessor performs steps (280-286) shown in Fig. 6, for one frame, which the image processor performs steps (290-298) for another frame (said ***parallel processing***) [col. 15, lines 23-32].

In regards to claim **12**, Bowen teaches a hyperpipe router (201) [Fig. 2], which determines which packet is intend for which pipe. The packed is then routed to a local router (202) that directs the packet to the appropriate circuit within pipe (e.g., the rasterizer) [col. 5 lines 46-60]. The rationale for combining as applied to claim 1 is incorporated herein.

In regards to claim **13**, as shown in Fig. 3 of the block diagram showing the relationship between hardware and software, the graphics support software (160) executes on the host computer system (130) and communicates with the image processing board (164) through the hardware abstraction layer (162) (said ***interface***) [col. 11, lines 3-14].

In regards to claim **14**, claim 14 recites the same limitations as claim 2. Therefore, the same rationale used for claim 2 is applied. Furthermore, the graphics support software (160) includes functions to support chunking and gsprite allocation [col. 11, lines 21-40]. Therefore, although not explicitly taught, it is implicitly that this information provided from the graphics software is then communicated to the image processing board (164) via the hardware abstraction layer (162).

In regards to claim **15**, claim 15 recites the same limitations as claim 3. Therefore, the same rationale used for claim 3 is applied.

In regards to claim **16**, claim 16 recites the same limitations as claim 4. Therefore, the same rationale used for claim 4 is applied.

In regards to claim **17**, claim 17 recites the same limitations as claim 5. Therefore, the same rationale used for claim 5 is applied.

In regards to claim **18**, Kenworthy does not explicitly disclose a bounding region calculator, however, the graphics support software (160) includes functions to support chunking and gsprite allocation, which therefore determines the bounding region for the geometry chunk [col. 11, lines 21-40]. Furthermore, Bowen teaches a hyperpipe router (201) [Fig. 2], which determines which packet is intend for which pipe. The packed is then routed to a local router (202) that directs the packet to the appropriate circuit within

pipe (e.g., the rasterizer) [col. 5 lines 46-60]. The rationale for combining as applied to claim 1 is incorporated herein.

In regards to claim **19**, claim 19 recites the same limitations as claim 8. Therefore, the same rationale used for claim 8 is applied.

In regards to claim **20**, claim 20 recites the same limitations as claim 9. Therefore, the same rationale used for claim 9 is applied.

In regards to claim **21**, claim 21 recites the same limitations as claim 10. Therefore, the same rationale used for claim 10 is applied.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michelle K. Lay whose telephone number is (571) 272-7661. The examiner can normally be reached on Monday-Friday 7:30a-5p.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kee M. Tung can be reached on (571) 272-7794. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Michelle K. Lay
Patent Examiner
Division 2628
08.07.2007 mkl

Michelle K. Lay
Michelle K. Lay
Patent Examiner

KEE M. TUNG
SUPERVISORY PATENT EXAMINER